Amendments to the Claims:

1. (currently amended) A processing system electrically connected to a computer, the processing system comprising:

a non-volatile memory (NVM) for storing firmware needed by the processing system; and

an NVM control interface having a plurality of registers for updating and reading data stored in the NVM;

wherein when the NVM control interface updates a current piece of data stored in the NVM, the NVM control interface is for first ~~reads~~ reading a ~~prior~~ previous piece of data ~~that is being already stored~~ updated in the NVM prior to the current piece of data and transmits the ~~prior~~ previous piece of data to the computer for comparison with original data that was previously stored in the NVM, if the previous piece of data and the original data are identical, ~~then~~ the NVM control interface updates the current piece of data.

2. (original) The processing system of claim 1 further comprising a processor for controlling operations of the processing system and a bus controller electrically connected to the processor, the NVM, and the NVM control interface for controlling data transmission between the processor, the NVM, and the NVM control interface.

3. (currently amended) The processing system of claim 2 wherein the processor is ~~capable of~~for loading the firmware from the NVM via the bus controller.

4. (original) The processing system of claim 1 further comprising a serial port interface electrically connected between the computer system and the NVM control interface for converting serial data bits received from the computer into data bytes and for converting data bytes received from the NVM control interface into serial data bits.

5. (original) The processing system of claim 4 wherein a serial port of the serial port interface is a RS-232 (Recommended Standard-232) port.

6. (currently amended) The processing system of claim 1 wherein before the NVM control interface reads the previous~~prior~~ piece of data, the NVM control interface verifies ~~that the~~whether or not the previous~~prior~~ piece of data has been written into the NVM.

7. (original) The processing system of claim 1 wherein the NVM is a flash read-only memory (flash ROM).

8. (currently amended) The processing system of claim 1 wherein the NVM control interface comprises:

an NVM address register for ~~setting~~ specifying addresses of the NVM, wherein an address stored in NVM address register is ~~increased~~ incremented after a write/read (W/R) operation of the NVM is finished;

an NVM page register for ~~setting~~ specifying a download capacity of the NVM;

an NVM data register for storing a data byte stored in the address ~~set~~ specified by the NVM address register;

a plurality of control bits for setting an operational mode of the NVM;

and

a plurality of command registers for ~~executing~~specifying commands so as to control operations of the NVM.

9. (currently amended) The processing system of ~~claim 1~~ claim 2 wherein during the update of the firmware stored in the NVM, the bus controller is for preventing ~~no~~ data access between the processor and the NVM ~~is allowed~~.

10. (currently amended) A method for updating a firmware stored in a non-volatile memory (NVM) of a processing system, the processing system being electrically connected to a computer system and further comprising a NVM control interface having a plurality of registers for updating and accessing data stored in the NVM, the method comprising:

~~using the NVM control interface to update~~ updating a current piece of data stored in the NVM utilizing the NVM control interface; and

during the update of the current piece of data, ~~using the NVM control interface to read~~ reading a previous ~~prior~~ piece of data ~~that is~~ being stored in the NVM prior to the current piece of data and transmitting the previous ~~prior~~ piece of data to the computer for comparison with original data that was previously stored in the NVM, if the previous piece of data and the original data are identical, then updating the current piece of data utilizing the NVM control interface.

11. (currently amended) The method of claim 10 wherein the processing

system further comprises a processor for controlling operations of the processing system and a bus controller electrically connected to the processor, the NVM, and the NVM control interface for controlling ~~datatramsission~~ data transmission between the processor, the NVM, and the NVM control interface.

12. (currently amended) The method of claim 11 wherein the processor is ~~capable of~~for loading the firmware stored in the NVM via the bus controller.

13. (original) The method of claim 10 wherein the processing system further comprises a serial port interface electrically connected between the computer system and the NVM control interface for converting serial data bits received from the computer into data bytes and for converting data bytes received from the NVM control interface into serial data bits.

14. (original) The method of claim 13 wherein a serial port of the serial port interface is a RS-232 (Recommended Standard-232) port.

15. (currently amended) The method of claim 10 further comprising:

verifying ~~that~~ whether or not the previous~~prior~~ piece of data has been written into the NVM before the NVM control interface reads the previous~~prior~~ piece of data.

16. (original) The method of claim 10 wherein the NVM is a flash read-only memory (flash ROM).

17. (currently amended) The method of claim 10 wherein the NVM control interface comprises:

an NVM address register for ~~setting~~ specifying addresses of the NVM, wherein an address stored in NVM address register is ~~increased~~ incremented after a write/read (W/R) operation of the NVM is finished;

an NVM page register for ~~setting~~ specifying a download capacity of the NVM;

an NVM data register for storing a data byte stored in the address ~~set~~ specified by the NVM address register;

a plurality of control bits for ~~setting~~ specifying an operation mode of the NVM; and

a plurality of command registers for ~~executing~~ specifying commands so as to control operations of the NVM.

18. (currently amended) The method of ~~claim 10~~ claim 11 further comprising:

~~forbidding~~ preventing any data access between the processor and the NVM during the update of the firmware stored in the NVM by utilizing the bus controller.